

---

# Random Maxout Features

---

**Youssef Mroueh Steven Rennie Vaibhava Goel**  
 IBM T.J Watson Research Center  
 {mroueh, sjrennie, vgoel}@us.ibm.com

## Abstract

In this paper, we propose and study random maxout features, which are constructed by first projecting the input data onto sets of randomly generated vectors with Gaussian elements, and then outputting the maximum projection value for each set. We show that the resulting random feature map, when used in conjunction with linear models, allows for the locally linear estimation of the function of interest in classification tasks, and for the locally linear embedding of points when used for dimensionality reduction or data visualization. We derive generalization bounds for learning that assess the error in approximating locally linear functions by linear functions in the maxout feature space, and empirically evaluate the efficacy of the approach on the MNIST and TIMIT classification tasks.

## 1 Introduction

Kernel based learning algorithms are ubiquitous in both supervised and unsupervised learning. For example, a universal kernel support vector machine approximates, to an arbitrary precision, any non-linear decision boundary function given enough training points [1]. On the other hand, methods like Kernel Principal Component Analysis [2] (Kernel PCA) capture non-linear relationships between variables of interest, and are used in non-linear dimensionality reduction. However, non-linear kernel methods suffer from high computational complexity (often cubic in the sample size), and are difficult to parallelize—training and testing on a even modestly sized dataset such as the TIMIT speech corpus (2M training samples) can be very challenging. Linear methods, on the other hand (linear support vector machines, logistic regression, ridge regression, Principal Component Analysis, etc. ), suffer from low capacity and representation power, but have computational complexity linear in the sample size, and so can be more readily scaled to large data corpora. Scalability and non-linear representation power are two desiderata of any learning algorithm. Deep Neural Networks owe their success to this property, as they allow rich, non-linear modeling and they scale linearly in the sample size when trained with variants of stochastic gradient descent [3].

**Kernel Approximation with Random Features.** An elegant approach to overcoming the computational load of kernel methods, pioneered by [4], consists of generating explicit, randomized feature maps  $\Phi : \mathcal{X} \subset \mathbb{R}^d \rightarrow \mathbb{R}^m$ , where  $m$  is typically larger than the dimension of the input space  $n$ , to approximate the kernel  $K$ :

$$\text{For } (x, z) \in \mathcal{X}, \quad K(x, z) \approx \langle \Phi(x), \Phi(z) \rangle. \quad (1)$$

When used in conjunction with linear methods, such randomized features reveal the non-linear structure in the data, and we gain scalability, as linear methods scale linearly with training sample size. Random Fourier features, introduced in [4], approximate shift invariant kernels. For example, the Gaussian kernel,  $K(x, z) = \exp\left(-\frac{\|x-z\|^2}{2\sigma^2}\right)$ , can be approximated using the following feature map:

$$\Phi(x) = (\cos(w_1^\top x + b_1) \dots \cos(w_m^\top x + b_m)),$$

where  $w_i \sim \mathcal{N}(0, \frac{1}{\sigma^2} I_d)$ , are independent gaussian vectors, and  $b_i$  are independently and uniformly drawn from  $[0, 2\pi]$ . Recently [5] showed that a highly oversampled random Fourier features map

( $m = 400K$ ), and a large scale linear least squares classifier, approaches the performance of dense deep neural networks trained on the TIMIT speech corpus.

**Learning with Random Features.** More formally in a classical supervised learning setting, let  $\mathcal{X} \subset \mathbb{R}^d$ , be the input space and  $\mathcal{Y} = \{-1, 1\}$  be the label space in a binary classification setting. We are given a training set  $S = \{(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}, i = 1 \dots N\}$ . For kernel methods, the goal is to find a non-linear function  $f$  mapping  $\mathcal{X}$  to  $\mathcal{Y}$ , given a certain measure of discrepancy or a loss function  $V$ . The function  $f$  is restricted to belong to a hypothesis class of functions  $\mathcal{H}_K$ , the so called reproducing kernel Hilbert space (RKHS). Empirical risk minimization in that setup leads to a rich class of non-linear algorithms, via regularization in RKHS [6],

$$\min_{f \in \mathcal{H}_K} \frac{1}{N} \sum_{i=1}^N V(y_i, f(x_i)) + \lambda \|f\|_{\mathcal{H}_K}^2, \quad (2)$$

where  $\lambda > 0$  is the regularization parameter and  $\|f\|_{\mathcal{H}_K}$  is the norm in the RKHS. The optimum  $f^*$  of the problem in (2) has the following form  $f^*(x) = \sum_{i=1}^N \beta_i^* K(x, x_i)$ ,  $\beta^* \in \mathbb{R}^N$ . Solving for  $\beta^*$  may have a computational complexity of  $O(N^3)$  (Regularized Least squares) or  $O(N^2)$  (Support Vector Machines). Using an explicit feature map  $\Phi$  that approximates such a kernel in conjunction with a linear model, it is therefore sufficient to estimate a scalable regularized linear model with a computational complexity linear in the number of training examples,

$$\min_{\alpha \in \mathbb{R}^m} \frac{1}{N} \sum_{i=1}^N V(y_i, \langle \alpha, \Phi(x_i) \rangle) + \lambda \|\alpha\|^2, \quad (3)$$

where  $\alpha^*$  is the optimal solution. For sufficiently large  $m$  we have:  $f^*(x) \approx \langle \alpha^*, \Phi(x) \rangle$ , and  $\alpha^*$  can be found in  $O(Nm)$  time using stochastic gradient descent. Similar ideas extend to the unsupervised case. Recently [7] introduced the randomized non-linear component analysis, where it is shown that Kernel PCA can be approximated by using random Fourier features followed by a linear PCA.

**Contributions.** In this paper, inspired by the recently introduced maxout network [8], we introduce a simple but effective non-linear random feature map, called random maxout features, that approximates functions of interest with piecewise-linear functions. Locally linear boundaries and components are interesting as they carry locally the linear structure in the data, and have the advantage of being interpretable in the original feature space of the data, but how should such a kernel method be formulated? In principle, such a mapping could be achieved via any locally linear kernel of the form  $K(x, z) = \langle x, z \rangle \kappa_\sigma(x, z)$ , where  $\kappa_\sigma$  is a localizing kernel, such as, for example a Gaussian kernel with bandwidth  $\sigma$ . However, how to efficiently realize such a conditionally linear kernel is not clear; for example, achieving this via random Fourier features would involve taking the Kronecker product of a linear feature map and the random features. The main contribution of this paper is to introduce and analyze the random maxout feature map, which has the advantage that it can be learned in  $O(Nm)$  time ( $N$  training points,  $m$  random features), and utilized at test time in  $O(dm)$  time (assuming  $d$  input features), while avoiding the taking Kronecker products. When used in conjunction with linear methods, random maxout realize a scalable, local linear function estimator for large-scale classification and regression. In the unsupervised setting, similarly to [7], random maxout features followed by a PCA allow for a locally linear embedding of the data, that can be used as a non-linear dimensionality reduction, and for data visualization. The paper is organized as follows: In Section 2 we introduce our random maxout feature map, and show that its expected kernel is indeed locally linear. In section 3 we present generalization bounds for the learning of linear functions in the random maxout feature space. In section 4, we discuss how random maxout features relate to previous work. Finally, in section 5, we demonstrate the approach as a local linear estimator in a classification setting on MNIST and TIMIT speech corpora.

## 2 Random Maxout Features

Random maxout features have the same structure as deep maxout networks in terms of maxout units. The following definition gives a precise description of the maxout random feature map:

**Definition 1** (Random Maxout Features). Let  $w_j^\ell$ ,  $\ell = 1 \dots m$ , and  $j = 1 \dots q$ , be independent random gaussian vectors i.e  $w_j^\ell \sim \mathcal{N}(0, I_d)$ . Note  $W^\ell = (w_1^\ell \dots w_q^\ell)$

For  $x \in \mathbb{R}^d$ , we define a maxout random unit  $h_\ell(x)$  as follows:

$$h_\ell(x) = \phi(x, W^\ell) = \max_{j=1 \dots q} \langle w_j^\ell, x \rangle, \quad \ell = 1 \dots m.$$

A maxout random feature map  $\Phi$  is therefore defined as follows:

$$\Phi(x) = \frac{1}{\sqrt{m}}(h_1(x), \dots, h_m(x)).$$

In order to study this map, we shall consider, for 2 points  $x, z \in \mathbb{R}^d$ , the dot product:  $\langle \Phi(x), \Phi(z) \rangle = \frac{1}{m} \sum_{\ell=1}^m h_\ell(x) h_\ell(z)$ . Consider first the expectation of  $\langle \Phi(x), \Phi(z) \rangle$ :  $\mathbb{E}(\langle \Phi(x), \Phi(z) \rangle) = \frac{1}{m} \sum_{\ell=1}^m \mathbb{E}(h_\ell(x) h_\ell(z)) = \mathbb{E}(h_1(x) h_1(z))$ , where the last equality follows from the independence of the units. It is therefore sufficient to study the expectation of the dot product of one unit:

$$K(x, z) = \mathbb{E}(h(x) h(z)),$$

where  $h(x) = \max_{j=1 \dots q} \langle w_j, x \rangle$ ,  $w_j \sim \mathcal{N}(0, I_d)$ ,  $j = 1 \dots q$ , iids.

**Theorem 1** (Maxout Expected Kernel). *Let  $x, z \in \mathbb{R}^d$ . The expected kernel of maxout random units is given by the following expression:*

$$K(x, z) = \mathbb{E}(h(x) h(z)) = \sigma^2(q) \langle x, z \rangle \kappa_q(x, z),$$

where  $\sigma^2(q) = \mathbb{E}(\max_{j=1 \dots q} g_j)^2$ ,  $g_j \sim \mathcal{N}(0, 1)$  iid, and  $\kappa_q(x, z)$  is a non-linear kernel given by:

$$\kappa_q(x, z) = \mathbb{P} \left\{ \arg \max_{j=1 \dots q} \langle w_j, x \rangle = \arg \max_{j=1 \dots q} \langle w_j, z \rangle \right\} = \sum_{i=0}^{\infty} a_i(q) \left( \frac{\langle x, z \rangle}{\|x\| \|z\|} \right)^i,$$

where the first 3 coefficients are  $a_0(q) = \frac{1}{q}$ ,  $a_1(q) = \frac{h_1^2(q)}{q-1}$ ,  $a_2(q) = \frac{qh_2^2(q)}{(q-1)(q-2)}$ , where  $h_i(q) = \mathbb{E} \phi_i(\max_{k=1 \dots q} g_k)$ , where  $g_j, j = 1 \dots q$  are iid standard centered gaussian, and  $\phi_i$ , the normalized Hermite polynomials.  $a_i(q)$  are non negative and  $\sum_{i \geq 0} a_i(q) = 1$ .

*Proof of Theorem 1.* The proof is given in Appendix A in the supplementary material.  $\square$

## 2.1 Discussion of the Derived Maxout Kernel

The expected kernel of a maxout unit is therefore a locally weighted linear kernel, and hence it allows a non-linear estimation of functions in a piecewise linear way :

$$K(x, z) = \sigma^2(q) \langle x, z \rangle \kappa_q(x, z),$$

where  $\kappa_q$  is a non-linear kernel. Let  $\rho = \langle x, z \rangle$ . In this section we discuss the locality introduced by  $\kappa_q(\cdot, \cdot)$ .

It is important to note that  $0 \leq \kappa_q(x, z) \leq 1$ , since  $\kappa_q(x, z) = \mathbb{P}(D(x) = D(z))$ , where  $D(x) = \arg \max_{j=1 \dots q} \langle w_j, x \rangle$ . For simplicity assume that  $x, z \in \mathbb{S}^{d-1}$ , where  $\mathbb{S}^{d-1}$  is the unit sphere in  $d$  dimensions. We start by giving values of  $\kappa_q$  in three particular cases of interest:

1. When  $x$  and  $z$  coincide i.e  $x = z$ , and  $\rho = 1$ , we have  $\kappa_q(x, z) = 1$ , as  $\sum_{i \geq 0} a_i(q) = 1$ .
2. When  $x$  and  $z$  are orthogonal i.e  $\rho = 0$ , we have  $\kappa_q(x, z) = a_0(q) = \frac{1}{q}$ .
3. When  $x$  and  $z$  are diametrically opposed i.e  $x = -z$ , and  $\rho = -1$ , we have  $\kappa_q(x, z) = 0$ , as  $\sum_{i \geq 0} a_{2i}(q) = \sum_{i \geq 0} a_{2i+1}(q) = \frac{1}{2}$  [26].

In order to understand the locality introduced by the non-linear kernel  $\kappa_q$ , and the relation of the radius of the locality to the size of the pool  $q$ , looking to the first order expansion of  $\kappa_q$  gives us a hint on the effect of that kernel. In particular the quantity  $h_1(q)$  is just the expectation of the maximum of independent gaussians  $h_1(q) \sim \sqrt{2 \log(q)}$  [10, 26].

$$\begin{aligned} \kappa_q(x, z) &= a_0(q) + a_1(q)\rho + O(\rho^2) \\ &= \frac{1}{q} (1 + (1 + \epsilon(q))2 \log(q)\rho) + O(\rho^2), \end{aligned}$$

where  $\epsilon(q) \rightarrow 0$ , for  $q \rightarrow \infty$ .

Note by  $g$  the function,  $g : [-1, 1] \rightarrow [0, 1]$  such that  $\kappa_q(x, z) = g(\rho)$ . For far apart points, when  $\rho \rightarrow -1$ ,  $g(\rho) \rightarrow 0$ .  $g$  has a linear behavior around  $\rho = 0$ , with a slope equal to  $\frac{2 \log(q)}{q}$ . Note that in this neighborhood as  $q$  increases the linear regime vanishes, and  $g(\rho) \rightarrow 0$ . Hence as  $q$  increases the probability of two points hashing to same index of maximum becomes smaller; qualitatively the radius of the locality of  $\kappa_q$  shrinks as  $q$  increases. Finally for near by points when  $\rho \rightarrow 1$ ,  $g(\rho) \rightarrow 1$ . Qualitatively the derived kernel  $K(x, z) \approx 0$  for far apart points and  $K(x, z) \approx \sigma^2(q) \langle x, z \rangle$  for points in the same neighborhood, where the radius of the locality, and the notion of closeness is set by the choice of the size of the pool  $q$ . This radius is decreasing in  $q$ . Hence  $K$  defines a locally linear kernel.

Now if we go back to problem (2), and solve for  $f$  in the reproducing kernel hilbert space of the equivalent kernel  $K$  (i.e for  $\mathcal{H} = \mathcal{H}_K$ ), we have :

$$f^*(x) = \sum_{i=1}^N \beta_i^* K(x, x_i) = \sigma^2(q) \sum_{i=1}^N \beta_i^* \langle x, x_i \rangle \kappa_q(x, x_i) = \sigma^2(q) \left\langle \sum_{i=1}^N \beta_i^* \kappa_q(x, x_i) x_i, x \right\rangle. \quad (4)$$

Hence we see that this derived kernel allows a locally linear estimation of the function of interest  $f^*$ , where the radius of the locality is set by the choice of the size of the pool  $q$ .

Now consider the maxout random feature map  $\Phi$  introduced in Definition 1. Recall that we have:

$$\mathbb{E}(\langle \Phi(x), \Phi(z) \rangle) = K(x, z) = \sigma^2(q) \langle x, z \rangle \kappa_q(x, z),$$

the dot product  $\langle \Phi(x), \Phi(z) \rangle$  is therefore an estimator of  $K(x, z)$  i.e for sufficiently large  $m$ ,  $\langle \Phi(x), \Phi(z) \rangle \approx K(x, z)$ , hence we can use the feature map  $\Phi$ , and a simple linear model as in equation (3), and use the optimal weight  $\alpha^*$  to get an estimate of the locally linear estimation  $f^*$  produced by the derived kernel as in equation (4), i.e. we have for sufficiently large  $m$ :

$$\langle \alpha^*, \Phi(x) \rangle \approx f^*(x). \quad (5)$$

In the next section we analyze the errors incurred by such approximation and how it translates to the convergence of the risk to its optimal value in a dense subset of the RKHS induced by the locally linear kernel.

**Remark 1** (Locality Sensitive Hashing). Let  $C : \mathbb{S}^{d-1} \rightarrow \{1 \dots q\}^m$ , such that for  $x \in \mathbb{S}^{d-1}$ ,  $C(x) = (\arg \max_{j=1 \dots q} \langle w_j^1, x \rangle, \dots, \arg \max_{j=1 \dots q} \langle w_j^m, x \rangle)$ ,  $w_j^\ell \sim \mathcal{N}(0, I_d)$ ,  $\ell = 1 \dots m$ ,  $j = 1 \dots q$ . For  $x, z \in \mathbb{S}^{d-1}$  we have :  $\mathbb{E}(\frac{1}{m} \sum_{i=1}^m \mathbb{1}_{C_i(x) \neq C_i(z)}) = \mathbb{P}\{\arg \max_{j=1 \dots q} \langle w_j, x \rangle \neq \arg \max_{j=1 \dots q} \langle w_j, z \rangle\} = 1 - \kappa_q(x, z)$ . Hence we can approximate the local kernel  $\kappa_q(x, z)$ , by the non binary strings by mean of the hamming distance between the  $q$ -ary strings  $C(x)$ , and  $C(z)$ . As  $m$  becomes large we have:

$$d_H(C(x), C(z)) = \frac{1}{m} \sum_{i=1}^m \mathbb{1}_{C_i(x) \neq C_i(z)} \approx 1 - \kappa_q(x, z),$$

Hence  $C$  defines a locality sensitive hashing scheme in the sense of [11].

### 3 Learning with Random Maxout Features

We show in this section that learning a linear model in the random maxout feature space, allows for a locally linear estimation of functions in a supervised classification setting. The locally linear kernel  $K(x, z) = \sigma^2(q) \langle x, z \rangle \kappa_q(x, z)$ , defines a Reproducing Kernel Hilbert Space (RKHS). In the following we will see how linear functions in the random maxout feature space approximate a dense subset of this locally linear RKHS. We start by introducing some notation. We assume that we are given a training set  $S = \{(x_i, y_i), x_i \in \mathcal{M} = \mathcal{X} \cap \mathbb{S}^{d-1}, y_i \in \mathcal{Y} = \{-1, 1\}, i = 1 \dots N\}$ . Our goal is to learn a function  $f : \mathcal{M} \rightarrow \mathbb{R}$  via risk minimization. Let  $\rho_y(x)$  be the label posteriors and assume  $\mathcal{M}$  is endowed with a measure  $\rho_{\mathcal{M}}$ , the expected and empirical risks induced by a  $L$ -Lipchitz loss function  $V : \mathbb{R} \rightarrow [0, 1]$  are the following:

$$\mathcal{E}_V(f) = \int_{\mathcal{M}} \sum_{y \in \mathcal{Y}} V(yf(x)) \rho_y(x) d\rho_{\mathcal{M}}(x), \quad \hat{\mathcal{E}}_V(f) = \frac{1}{N} \sum_{i=1}^N V(y_i f(x_i)).$$

The assumptions on the points belonging to the unit sphere, and on the loss being bounded by one can be weakened see Remark 2. We will use in the following a notion of intrinsic dimension for the set  $\mathcal{M}$ , namely the Assouad dimension given in the following definition:

**Definition 2** ([12]). *The Assouad dimension of  $\mathcal{M} \subset \mathbb{R}^d$ , denoted by  $d_{\mathcal{M}}$ , is the smallest integer  $k$ , such that, for any ball  $B \subset \mathbb{R}^d$ , the set  $B \cap \mathcal{M}$  can be covered by  $2^k$  balls of half the radius of  $B$ .*

The Assouad dimension is used as a measure of the intrinsic dimension. For example, if  $\mathcal{M}$  is an  $\ell_p$  ball in  $\mathbb{R}^d$ , then  $d_{\mathcal{M}} = O(d)$ . If  $\mathcal{M}$  is a  $r$ -dimensional hyperplane in  $\mathbb{R}^r$ , then  $d_{\mathcal{M}} = O(r)$ , where  $r < d$ . Moreover, if  $\mathcal{M}$  is a  $r$ -dimensional Riemannian submanifold of  $\mathbb{R}^d$  with suitably bounded curvature, then  $d_{\mathcal{M}} = O(r)$ .

Let  $W^\ell = (w_1^\ell, \dots, w_q^\ell)$ ,  $\ell = 1 \dots m$ , and  $W = (w_1, \dots, w_q)$ , since  $w_j, j = 1 \dots q$  are iid, the distribution of  $W$  is given by  $p(W) = p(w_1) \dots p(w_q)$ , where  $p(w_j)$  is the distribution of a gaussian vector drawn from  $\mathcal{N}(0, I_d)$ . Similarly to the analysis in [13], let  $C > 0$ , we define the infinite dimensional functional space  $\mathcal{F}$ :

$$\mathcal{F} = \left\{ f(x) = \int \alpha(W) \phi(x, W) dW, \sup_W \frac{|\alpha(W)|}{p(W)} \leq C \right\},$$

it is easy to see that  $\mathcal{F}$  is dense in  $\mathcal{H}_K$  [13]. We will approximate the set  $\mathcal{F}$  with  $\hat{\mathcal{F}}$  defined as follows:

$$\hat{\mathcal{F}} = \left\{ f(x) = \sqrt{m} \langle \alpha, \Phi(x) \rangle = \sum_{\ell=1}^m \alpha_\ell \phi(x, W^\ell), \|\alpha\|_\infty \leq \frac{C}{m} \right\}.$$

Note that in this definition of this function space we are regularizing the norm infinity of the weight vectors this can be replaced in practice, and theory [14] by a classical Tikhonov regularization or other forms of regularization.

**Theorem 2** (Learning with Random Maxout Features). *Let  $S = \{(x_i, y_i), x_i \in \mathcal{M} = \mathcal{X} \cap \mathbb{S}^{d-1}, y_i \in \{-1, 1\}, i = 1 \dots N\}$ , and  $d_{\mathcal{M}}$  the assouad dimension of  $\mathcal{M}$ , and  $\text{diam}(\mathcal{M})$  be its diameter. Let  $\hat{f}_N = \arg \min_{f \in \hat{\mathcal{F}}} \hat{\mathcal{E}}_V(f)$ . Fix  $\delta > 0$ ,  $\varepsilon \in (0, 1)$ , for  $m \geq \frac{C'}{\varepsilon^2} \left( d_{\mathcal{M}} \log \left( \frac{\text{diam}(\mathcal{M}) \sqrt{d}}{\delta} \right) + \log(q+1) \right)$ , where  $C'$  is a numerical constant, we have:*

$$\mathcal{E}_V(\hat{f}_N) - \min_{f \in \hat{\mathcal{F}}} \mathcal{E}_V(\hat{f}) \leq 4LC \sqrt{\frac{\sigma^2(q)}{N}} + \frac{2|V(0)|}{\sqrt{N}} + 2\sqrt{\frac{2\log(1/\delta)}{N}} + LC\varepsilon \left( 1 + \sqrt{2\log\left(\frac{1}{\delta}\right)} \right),$$

with probability at least  $1 - 3\delta - 2e^{-cd/4}$ , on the choices of the training examples and the random projections. Where  $C$  is the regularization parameter in the definition of  $\mathcal{F}$  and  $\hat{\mathcal{F}}$ ,  $L$  is the lipchitz constant of the loss function  $V : \mathbb{R} \rightarrow [0, 1]$ , and  $\sigma^2(q) = \mathbb{E}(\max_{j=1 \dots q} g_j)^2$ ,  $g_j \sim \mathcal{N}(0, 1)$  iids.

The proof of Theorem 2 is given in the supplementary material in Appendix B, the main technical difficulty consists in bounding the  $\sup_{x \in \mathcal{M}} |\phi(x, W)|$ , and relating this quantity to the intrinsic dimension  $d_{\mathcal{M}}$ . Theorem 2, shows that for  $q > 1$ , learning a linear model in the maxout feature space defined by the map  $\Phi$  has a low expected risk and more importantly this risk is not far from the one achieved by a nonlinear infinite dimensional function class  $\mathcal{F}$ . Locally linear functions can be hence estimated to an arbitrary precision using linear models in the maxout feature space, the errors decompose naturally to an estimation or a statistical error with the usual rate of  $O(\frac{1}{\sqrt{N}})$ , and an approximation error of functions in the infinite dimensional space  $\mathcal{F}$ , by functions in  $\hat{\mathcal{F}}$ . For a fixed  $q$ , in order to achieve an approximation error  $\varepsilon$ , the bounds suggests  $\varepsilon = \frac{1}{\sqrt{N}}$ . One needs to set the dimensionality  $m$  of the feature map  $\Phi$  to  $O(N(d_{\mathcal{M}} \log(d) + \log(q+1)))$ , where  $d_{\mathcal{M}}$  is a measure of the intrinsic dimension of the space where inputs live  $d_{\mathcal{M}} \leq d$ . For instance if our data lived on a  $r$ -dimensional Riemannian submanifold of  $\mathbb{R}^d$  ( $r \ll d$ ), the function space  $\hat{\mathcal{F}}$  for  $m = O(N(r \log(d) + \log(q)))$  achieves an approximation error  $\varepsilon = \frac{1}{\sqrt{N}}$  of a dense subset of the function space defined by the local linear kernel, with a radius of locality set by the choice of the parameter  $q$ . As  $q$  increases this radius shrinks and the dimension of the feature map increases to ensure more locality but with a logarithmic dependency on  $q$ . The use of the intrinsic dimension of the inputs space  $\mathcal{M}$  -that we borrow from the compressive sensing community- in the approximation error is appealing as most of previous bounds in random features analysis relates the number

of projections only to the training size  $N$ , and spectral properties of the kernel matrix [14],[13]. Using spectral properties of the kernel, results in [14] suggest that for large  $N$  that the number of the features is of the order  $O(N \log(N))$ . While the spectral properties of the kernel carry some geometric information about the points distribution it misses some important geometric structure in the points set  $\mathcal{M}$ , since it captures some intrinsic dimension of the data that can be expressed only in term of the sample size  $N$ , while the Assouad dimension has a richer description of the structure in the data, such as sparsity for instance. If  $\mathcal{X}$  was the set of  $s$ -sparse signal the Assouad dimension  $d_{\mathcal{M}} = O(s \log(d))$  [12], and we need  $O\left(\frac{s \log^2(d) + \log(q)}{\varepsilon^2}\right)$  maxout random features to have an approximation error of  $\varepsilon$ . It would be interesting to incorporate in the bound both the spectral properties of the kernel and the intrinsic dimension to get the good of the two worlds, we leave this for a future work. For  $q = 1$ , Maxout random features reduces to classical random projections, that approximate the linear kernel, learning classifiers from randomly projected data has been thoroughly studied see [15], and references there in, Theorem 2 is not as sharp as results presented in [15], since the proof was not specialized to the linear projection case.

**Remark 2.** 1-We can relax the sphere constraint on the input set to a bounded data constraint, i.e  $\sup_{x \in \mathcal{X}} \|x\| \leq R$ , and assume a bounded loss  $|V(z)| \leq B$ , a minor change in the proof shows that the right hand side of the inequality in Theorem 2 is multiplied by  $RB$ .  
2-Note that for  $\varepsilon = \frac{1}{\sqrt{N}}$ , we have  $m = O(N d_{\mathcal{M}} \log(d))$ , for large  $N$  assume  $\mathcal{M}$  was finite and the cardinality  $|\mathcal{M}| = N^\alpha$  for small  $\alpha$ , we have  $d_{\mathcal{M}} = O(\log(|\mathcal{M}|)) = O(\log(N))$  and  $m = O(N \log(N) \log(d))$ , which matches up to a log term results in [14].

## 4 Related Work

**Approximating Kernels, Random Non Linear Embeddings.** The so called Johnson-Lindenstrauss Lemma [16] states that a linear random feature map preserves  $\ell_2$  distances in a  $N$ -point subset of a Euclidian space when embedded in  $O(\varepsilon^{-2} \log(N))$  dimension with a distortion of  $1 + \varepsilon$ . The requirement of preserving all pairwise distances is not needed in many applications; we need to preserve distances only in a local neighborhood of the points of interest. This observation is at the core of locality sensitive hashing [11] and has been discussed in [17]. One needs a non-linear random feature map in order to achieve a local embedding. Random Fourier features [4] approximating the Gaussian kernel achieve such a goal. Random maxout features also achieve such a goal by performing a locally linear embedding of the points.

**Scaling up Kernel Methods.** As discussed earlier random features is a popular approach in approximating the kernel matrix and scaling up kernel methods pioneered by [4], the generalization ability of such approach is of the order of  $\tilde{O}(\frac{1}{\sqrt{N}} + \frac{1}{\sqrt{m}})$ , which suggests that  $m$  needs to be  $\tilde{O}(N)$ . An elegant doubly stochastic gradient approach introduced recently in [18], uses random features to approximate the function space rather than the kernel matrix, in a memory efficient way that achieves this  $O(N)$  bound for the number of features, Maxout random features can be also used within this framework. Other approaches for scaling up kernel methods fall under the category of low rank Approximation of the kernel matrix, such as sparse greedy matrix approximation [19], Nystrom approximations [20] and low rank Cholesky decomposition [21].

**Locally Linear Estimation.** As discussed earlier, Random Maxout Networks allow us to do local linear estimation of functions in supervised and unsupervised learning tasks, among other approaches Deep Maxout Networks [8], Locally linear Embedding [22] and convex piecewise linear fitting [23], share similar structure with Random maxout features.

## 5 Numerical Experiments

### 5.1 Simulated Data Illustration

In this section we consider 100 points generated at random from the unit circle in two dimensions. We embed those points through the Maxout feature map  $\Phi$ , for  $m = 1000$ , and  $q = 2^3$  and  $q = 2^5$  respectively. We plot in Figure 1, for pairs of points  $x$  and  $z$ , the pairwise distances in the embedded space  $\|\Phi(x) - \Phi(z)\|$  versus the pairwise distances in the original  $\|x - z\|$  (we show here only a subset of those pairwise distances). We see that in both cases for small distances we have a linear regime, followed by a saturation regime for high range distances. The saturation arises earlier for

$q = 2^5$  when compared to the one of  $q = 2^3$ . This confirms the discussion in Section 2.1, on the effect of the Maxout feature map as a locally linear kernel, with the radius of the locality shrinking as  $q$  increases. To further illustrate this local linear embedding, we consider a dataset of 33 images

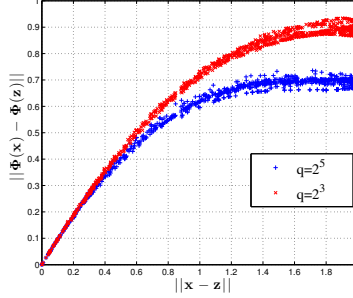


Figure 1: The locality induced by the size of the pool  $q$ . As  $q$  increase the locality radius shrinks.

of faces of the same person of size  $112 \times 92$  at different angles that we normalize to be unit norm (See Figure 2). The faces are ordered by their angles, the ordering provided in this dataset is noisy. We extract the maxout features on this dataset for  $m = 10000, q = 12$ , and perform principal component analysis on the data in the maxout feature space and project it down to two dimensions on the two largest principal components. We show in Figure 2, the embedding of this dataset in two dimensions through Maxout followed by a linear PCA. Each point in this scatter corresponds to a face, the numbering refers to the corresponding order in the given angle labeling. We see that Maxout local linear embedding self organizes the data with respect to the angle of variation, and corrects the noisy labeling.

## 5.2 Supervised Learning Applications: Classification

In order to perform classification as discussed earlier we lift the data through the maxout feature map  $\Phi$ , and solve a linear classification problem in the lifted space as described in Equation (3).

### 5.2.1 Digit Classification

We extract the Maxout random features on the MNIST dataset [3], consisting of 60000 training examples ( $d = 784$ ) and 10000 test examples among  $T = 10$  digits. Let  $Z = \Phi(X) \in \mathbb{R}^{N \times M}$  be the embedded data and  $Y \in \mathbb{R}^{N \times T}$  be the class labels using a  $\pm 1$  encoding. We solve the multi-class problem using a simple regularized least squares,  $f(x) = \langle \alpha, \Phi(x) \rangle$ , where  $\alpha = (Z^\top Z + \lambda I)^{-1} Z^\top Y$ , where  $\lambda$  is the regularization parameter chosen on a hold out validation set [30].

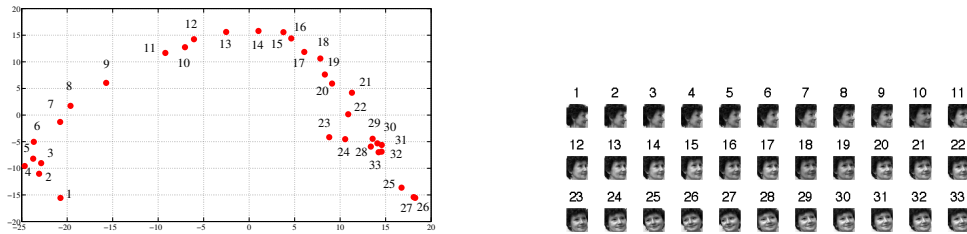


Figure 2: Maxout Locally Linear Embedding in 2D (Maxout - LLE) of 33 faces of the same person at different angles.

	$m = 100$	$m = 500$	$m = 1000$	$m = 5000$	$m = 10000$
$q = 1$	$17.68 \pm 0.28$	$14.76 \pm 0.23$	$14.64 \pm 0.08$	$15.02 \pm 0.64$	$15.12 \pm 0.58$
$q = 2$	$16.95 \pm 0.5321$	$7.98 \pm 0.22$	$5.62 \pm 0.18$	$2.86 \pm 0.11$	$2.35 \pm 0.04$
$q = 4$	$18.24 \pm 0.39$	$7.70 \pm 0.24$	$5.50 \pm 0.16$	$2.78 \pm 0.07$	<b><math>2.23 \pm 0.07</math></b>
$q = 8$	$18.9 \pm 0.28$	$7.98 \pm 0.47$	$5.59 \pm 0.19$	$2.81 \pm 0.10$	$2.32 \pm 0.09$
$q = 16$	$20.57 \pm 0.76$	$8.19 \pm 0.21$	$5.66 \pm 0.15$	$2.87 \pm 0.12$	$2.58 \pm 0.04$

Table 1: Random Maxout on MNIST: Error rates in %.

In table 1 we report test errors and standard deviations for various values of  $m$  and  $q$  in the maxout feature map averaged on 5 different choices of the random weights in the map. We see that for  $q = 1$ , where the feature map does not introduce any non linearity, the performance of the map for any value of  $m$ , matches the error rate of a linear classifier that is 15%. For  $q \neq 1$ , we start to see the non linearity introduced by the map as a local linear estimator, for a fixed  $q$  the error rate decreases as  $m$  gets large. In this experiment the best error rate is achieved for  $m = 10000$  and  $q = 4$ , suggesting that  $q = 4$  sets the optimal radius of locality for classification. As a baseline an optimal  $k$ -nearest neighbor achieves an error rate of 3.09 %.

### 5.2.2 Phone Classification on TIMIT

We further evaluated random maxout features on the TIMIT speech phone classification task. Evaluations are reported on the core test set of TIMIT. We utilized essentially the same experimental setup as in [5]: 147 context independent states were used as classification targets; at test time each utterance was decoded using the Viterbi algorithm, and then mapped, as is standard, down to 39 phones for scoring. As in [5], 2 million frames of training data—fMLLR features of dimension 40 each [24], spliced with  $\pm 5$  frames of context ( $d = 11 \times 40 = 440$ )—were utilized. These features were then lifted through the random maxout map  $\Phi$ , and a multinomial logistic regression was subsequently trained using SGD to minimize cross entropy loss. Table 2 reports the mean and standard deviation of the performance of random maxout units as a function of number of maxout features,  $m$ , and number of projections/feature,  $q$ . Interestingly, even smaller feature maps far outperform using the raw features, and the performance varies very little with initialization seed (5 seeds/result).

m	q			
	2	4	8	16
1250	$24.7 \pm 0.2$	$24.4 \pm 0.2$	$25.6 \pm 0.2$	$26.0 \pm 0.2$
2500	$24.0 \pm 0.2$	$23.3 \pm 0.3$	$24.7 \pm 0.3$	$25.3 \pm 0.3$
5000	$23.5 \pm 0.1$	$22.9 \pm 0.2$	$24.7 \pm 0.2$	$24.7 \pm 0.4$
10000	$23.2 \pm 0.1$	$22.5 \pm 0.2$	$24.5 \pm 0.2$	$24.7 \pm 0.4$
20000	$23.1 \pm 0.1$	$22.3 \pm 0.2$	$24.3 \pm 0.2$	$24.5 \pm 0.2$

Table 2: Phone error rate (PER, %) as a function of number of maxout features,  $m$ , and number of linear projections per maxout feature,  $q$ , on the TIMIT speech phone classification task. Multinomial logistic regression on the input features yields a PER of  $33.1 \pm 0.1\%$ .

Table 3 summarizes preliminary investigations into scaling up the size of the feature map, where to increase the number of features, projections are shared across random maxout units. Random maxout features appear to perform similarly to random Fourier features on the task.



network	# features (m)	#projections	#proj./feature (q)	phone error rate (PER)
Random Maxout	15K	15K	q=4	23.1
Random Maxout	60K	15K	q=4	22.7
Random Maxout	60K	60K	q=4	22.8
Random Maxout	400K	15K	q=4	22.4
Random Maxout	300K	300K	q=4	22.1
Random Fourier	400K	400K	-	21.3 [5]
ReLU DNN	4K,	16K (4Kx 4 layers)	-	22.7 [25]
ReLU DNN w/ dropout	4K,	16K (4Kx 4 layers)	-	19.7 [25]

Table 3: Phone error rates (PER,%) on TIMIT. The total number of projections used to produce each feature map are as indicated (here random maxout features draw from a shared pool of projections).

In this paper we presented random maxout feature map as an effective and scalable local linear estimator, and derived risk bounds for learning in this feature space that assesses both statistical and approximation errors, in a classification setting. We believe that maxout features, thanks to their conditionally linear structure, can gain further in scalability, and speed, by leveraging the fast Johnson Lindenstrauss transform, and the doubly stochastic optimization framework of [18].

## A Proof of Theorem 1

*Proof of Theorem 1.* Assume without loss of generality that  $\|x\| = \|z\| = 1$ . Let  $D(x) = \arg \max_{j=1\dots q} \langle w_j, x \rangle$ , and  $D(z) = \arg \max_{j=1\dots q} \langle w_j, z \rangle$ , ties are broken arbitrarily. By total probability we have:

$$\begin{aligned}
K(x, z) &= \mathbb{E}(h(x)h(z)) \\
&= \mathbb{E}\{h(x)h(z) | D(x) = D(z)\} \mathbb{P}(D(x) = D(z)) \\
&\quad + \mathbb{E}\{h(x)h(z) | D(x) \neq D(z)\} \mathbb{P}(D(x) \neq D(z)) \\
&= \mathbb{E}(\langle w_{D(x)}, x \rangle \langle w_{D(x)}, z \rangle | D(x) = D(z)) \mathbb{P}(D(x) = D(z)) \\
&\quad + \mathbb{E}\{\langle w_{D(x)}, x \rangle \langle w_{D(z)}, z \rangle | D(x) \neq D(z)\} \mathbb{P}(D(x) \neq D(z)).
\end{aligned}$$

It is easy to see that the second term in this sum is zero since the gaussians are independent and zero centered :  $\mathbb{E}\{\langle w_{D(x)}, x \rangle \langle w_{D(z)}, z \rangle | D(x) \neq D(z)\} = 0$ . We are left with the first term of this sum:

$$\begin{aligned}
K(x, z) &= \mathbb{E}(h(x)h(z)) \\
&= \mathbb{E}(\langle w_{D(x)}, x \rangle \langle w_{D(x)}, z \rangle | D(x) = D(z)) \mathbb{P}(D(x) = D(z)) \\
&= q \mathbb{E}(\langle w_1, x \rangle \langle w_1, z \rangle | D(x) = D(z) = 1) \mathbb{P}(D(x) = D(z) = 1) \\
&= \mathbb{E}(\langle w_1, x \rangle \langle w_1, z \rangle | D(x) = D(z) = 1) \mathbb{P}(D(x) = D(z)).
\end{aligned}$$

By rotation invariance of gaussians we have:

$$\langle w_1, x \rangle = g \text{ and } \langle w_1, z \rangle = \langle x, z \rangle g + \sqrt{1 - |\langle x, z \rangle|^2} h,$$

where  $g$  and  $h$  are independent random gaussian variables  $g, h \sim \mathcal{N}(0, 1)$ .

Let  $E$  be the following event :

$$E = \{g \text{ is the maximum of } q \text{ independent gaussians}\}$$

Hence we have:

$$\begin{aligned}
&\mathbb{E}(\langle w_1, x \rangle \langle w_1, z \rangle | D(x) = D(z) = 1) \\
&= \mathbb{E}\left(g(\langle x, z \rangle g + \sqrt{1 - |\langle x, z \rangle|^2} h) | E\right) \\
&= \langle x, z \rangle \mathbb{E}\left(\left[\max_{j=1\dots q} g_j\right]^2\right).
\end{aligned}$$

Let  $\sigma^2(q) = \mathbb{E} \left( [\max_{j=1 \dots q} g_j]^2 \right)$ , we have finally:

$$\mathbb{E}(h(x)h(z)) = \sigma^2(q) \langle x, z \rangle \mathbb{P}(D(x) = D(z)). \quad (6)$$

$\sigma^2(q)$  is a normalization factor and it is well known that  $\sigma^2(q) \sim \log(q)$ , hence we are left with

$$\mathbb{P}(D(x) = D(z)),$$

that is the probability that  $x$  and  $z$  are not separated by the  $q$  hyperplanes, an object that is well studied in  $q$ -ways graph cuts approximation algorithms.

The following lemma is crucial to our proof and is proved in [26], and allow us to get the final expression of the expected kernel.

**Lemma 1** ([26]). *For  $x, z \in \mathbb{R}^d$ ,  $\|x\| = \|z\| = 1$ . Let  $\rho = \langle x, z \rangle$ , we have therefore:*

$$\kappa_q(x, z) = \mathbb{P}(D(x) = D(z)) = \sum_{i=0}^{\infty} a_i(q) \rho^i \quad (7)$$

the taylor series of  $\kappa_q$  around  $\rho = 0$ , converges for all  $\rho$  in the range  $|\rho| \leq 1$ . The coefficients  $a_i(q)$ , of the expansion are all non negatives and their sum converges to 1. The first 3 coefficients are  $a_0(q) = \frac{1}{q}$ ,  $a_1(q) = \frac{h_1^2(q)}{q-1}$ ,  $a_2(q) = \frac{qh_2^2(q)}{(q-1)(q-2)}$ .  $h_i(q) = \mathbb{E}\phi_i(\max_{k=j \dots q} \eta_j)$ , where  $\eta_j, j = 1 \dots q$  are iid standard centered gaussian, and  $\phi_i$ , the normalized Hermite polynomials.

By lemma 1 we have finally:

$$K(x, z) = \mathbb{E}(h(x)h(z)) = \sigma^2(q) \langle x, z \rangle \kappa_q(x, z), \quad (8)$$

where  $\kappa_q(x, z) = \sum_{i=0}^{\infty} a_i(q) (\langle x, z \rangle)^i$ , is a non-linear kernel, values of  $a_i(q)$  are given in the above lemma.  $\square$

## B Learning with Random Maxout Features

In this section we state the proof of Theorem 2. We start with a preliminary Lemma that bounds  $\phi(x, W)$  uniformly on the set  $\mathcal{X}$ , this will be crucial in our derivations.

**Lemma 2** (Bounding  $\sup_{x \in \mathcal{M}} |\phi(x, W)|$ ). *Let  $\mathcal{M} = \mathcal{X} \cap \mathbb{S}^{d-1}$ . Let  $d_{\mathcal{M}}$  be the Assouad dimension of  $\mathcal{M}$  and  $\text{diam}(\mathcal{M})$  be the diameter of  $\mathcal{M}$ . Let  $\delta > 0$ , we have for a numeric constant  $C_1$ :*

$$\sup_{x \in \mathcal{M}} |\phi(x, W)| = \sup_{x \in \mathcal{M}} \left| \max_{j=1 \dots q} \langle w_j, x \rangle \right| \leq C_1 \sqrt{d_{\mathcal{M}} \log \left( \frac{\text{diam}(\mathcal{M}) \sqrt{d}}{\delta} \right) + \log(q+1)},$$

with probability at least  $1 - \delta - 2e^{-cd/4}$ .

*Proof.* Consider an  $\epsilon$ -Net that covers  $\mathcal{X}$  with balls of radius  $r$  and centers  $\{x_i\}_{i=1 \dots T}$ . We have by definition of the Assouad dimension, the maximum number of balls  $T$  is less than  $\left( \frac{2\text{diam}(\mathcal{M})}{r} \right)^{d_{\mathcal{M}}}$ . Assume we have:  $|\phi(x, W) - h(z, W)| = \langle w_{D(x)}, x \rangle - \langle w_{D(z)}, z \rangle$ , meaning  $\langle w_{D(x)}, x \rangle - \langle w_{D(z)}, z \rangle > 0$ .

$$\begin{aligned} \phi(x, W) - \phi(z, W) &= \langle w_{D(x)}, x \rangle - \langle w_{D(z)}, z \rangle \\ &= \langle w_{D(x)}, x - z \rangle \\ &\quad - \underbrace{\langle w_{D(z)} - w_{D(x)}, z \rangle}_{\geq 0} \\ &\leq \|w_{D(x)}\|_2 \|x - z\|_2, \end{aligned}$$

where the inequality follows from the definition of  $D(z)$ , and the Cauchy-Schwarz inequality. Similarly if we have  $|\phi(x, W) - \phi(z, W)| = \langle w_{D(z)}, z \rangle - \langle w_{D(x)}, x \rangle$ , we have:

$$\phi(x, W) - \phi(z, W) \leq \|w_{D(z)}\|_2 \|x - z\|_2$$

We conclude therefore that:

$$|\phi(x, W) - \phi(z, W)| \leq \max(\|w_{D(x)}\|, \|w_{D(z)}\|) \|x - z\|_2 \leq (\|w_{D(x)}\| + \|w_{D(z)}\|) \|x - z\|_2$$

Let  $L = \|w_{D(x)}\|_2 + \|w_{D(z)}\|_2$ .

Let  $t > 0$ , we have  $\sup_{x \in \mathcal{M}} |\phi(x, W)| < t$ , if the following two events hold:

$$E_1 = \left\{ \sup_{x_i, i=1 \dots T} |\phi(x_i, W)| < \frac{t}{2} \right\} \text{ and } E_2 = \left\{ L \leq \frac{t}{2r} \right\}.$$

On the first hand:

$$\begin{aligned} \mathbb{P}(E_1^c) &= \mathbb{P} \left( \sup_{x_i, i=1 \dots T} |\phi(x_i, W)| \geq \frac{t}{2} \right) \\ &= \mathbb{P} \left( \cup_{i=1}^T \{ |\phi(x_i, W)| \geq \frac{t}{2} \} \right) \\ &\leq \sum_{i=1}^T \mathbb{P} \left( |\phi(x_i, W)| \geq \frac{t}{2} \right) \\ &= T \mathbb{P} \left( \left| \max_{j=1 \dots q} \langle w_j, x \rangle \right| \geq \frac{t}{2} \right). \end{aligned}$$

Note that by a union bound we have:

$$\mathbb{P} \left( \max_{j=1 \dots q} \langle w_j, x \rangle \geq \frac{t}{2} \right) = \mathbb{P} \left( \exists j, \langle w_j, x \rangle \geq \frac{t}{2} \right) \leq q \mathbb{P}(\langle w, x \rangle \geq \frac{t}{2}) \leq q e^{-t^2/8},$$

and by independence of  $w_j$  we have also:

$$\mathbb{P} \left( \max_{j=1 \dots q} \langle w_j, x \rangle \leq -\frac{t}{2} \right) = \mathbb{P} \left( \forall j, \langle w_j, x \rangle \leq -\frac{t}{2} \right) = \left( \mathbb{P}(\langle w, x \rangle \leq -\frac{t}{2}) \right)^q \leq e^{-qt^2/8}.$$

Putting together theses to bounds we have:

$$\mathbb{P} \left( \left| \max_{j=1 \dots q} \langle w_j, x \rangle \right| \geq \frac{t}{2} \right) \leq q e^{-t^2/8} + e^{-qt^2/8}.$$

The covering number  $T$  of  $\mathcal{X}$  is also bounded as follows [27]:

$$T \leq \left( \frac{2 \text{diam}(\mathcal{M})}{r} \right)^{d_{\mathcal{M}}}.$$

Hence we have for  $q > 1$ :

$$\mathbb{P}(E_1^c) \leq \left( \frac{2 \text{diam}(\mathcal{M})}{r} \right)^{d_{\mathcal{M}}} (q e^{-t^2/8} + e^{-qt^2/8}) \leq \left( \frac{2 \text{diam}(\mathcal{M})}{r} \right)^{d_{\mathcal{M}}} (q+1) e^{-t^2/8}.$$

On the other hand, for a universal constant  $c$ , and for  $\varepsilon \in (0, 1)$  [28]:

$$\mathbb{P}(\|w\|_2 \geq \sqrt{d}(1 + \varepsilon)) \leq e^{-c\varepsilon^2 d}.$$

Set  $\frac{t}{2r} = \sqrt{d}(1 + \varepsilon)$ , hence  $\mathbb{P}(E_2^c) \leq 2e^{-c\varepsilon^2 d}$ .

It follows that for  $t > 1$ :

$$\begin{aligned} \mathbb{P}(\sup_{x \in \mathcal{M}} |\phi(x, W)| \geq t) &\leq \mathbb{P}(E_1^c \cup E_2^c) \\ &\leq \mathbb{P}(E_1^c) + \mathbb{P}(E_2^c) \\ &\leq \left( \frac{4\sqrt{d}(1 + \varepsilon) \text{diam}(\mathcal{M})}{t} \right)^{d_{\mathcal{M}}} (q+1) e^{-t^2/8} + 2e^{-c\varepsilon^2 d} \\ &\leq \left( 4\sqrt{d}(1 + \varepsilon) \text{diam}(\mathcal{M}) \right)^{d_{\mathcal{M}}} (q+1) e^{-t^2/8} + 2e^{-c\varepsilon^2 d}. \end{aligned}$$

Hence for  $\varepsilon = \frac{1}{2}, t > 1$ :

$$\sup_{x \in \mathcal{M}} |\phi(x, W)| \leq t,$$

with probability at least  $1 - \left(6 \text{diam}(\mathcal{M}) \sqrt{d}\right)^{d\mathcal{M}} (q+1)e^{-t^2/8} - 2e^{-cd/4}$ .

Hence we have for a numeric constant  $C_1$ :

$$\sup_{x \in \mathcal{M}} |\phi(x, W)| \leq C_1 \sqrt{d_{\mathcal{M}} \log \left( \frac{\text{diam}(\mathcal{M}) \sqrt{d}}{\delta} \right) + \log(q+1)},$$

with probability at least  $1 - \delta - 2e^{-cd/4}$ .  $\square$

The following Lemma shows that any function  $f \in \mathcal{F}$ , can be approximated by a function  $\hat{f} \in \hat{\mathcal{F}}$ :

**Lemma 3.** [Approximation Error.] Let  $f$  be a function in  $\mathcal{F}$ . Then for  $\delta > 0$ , there exists a function  $\hat{f} \in \hat{\mathcal{F}}$  such that:

$$\|\hat{f} - f\|_{\mathcal{L}^2(\mathcal{X}, \rho_{\mathcal{M}})} \leq CC_1 \sqrt{\frac{d_{\mathcal{M}} \log \left( \frac{\text{diam}(\mathcal{M}) \sqrt{d}}{\delta} \right) + \log(q+1)}{m}} \left( 1 + \sqrt{2 \log \left( \frac{1}{\delta} \right)} \right)$$

with probability at least  $1 - 2\delta - 2e^{-cd/4}$ .

*Proof of Lemma 3.* Let  $f \in \mathcal{F}, f(x) = \int \alpha(W) \phi(x, W) dW$ . Let  $f_{\ell}(x) = \frac{\alpha(W^{\ell})}{p(W^{\ell})} \phi(x, W^{\ell})$ . We have the following:  $\mathbb{E}_W(f_{\ell}) = f$ , and  $\frac{1}{m} \mathbb{E}_W(\sum_{\ell=1}^m f_{\ell}) = f$ . Consider the Hilbert space  $\mathcal{L}^2(\mathcal{X}, \rho_{\mathcal{M}})$ , with dot product:  $\langle f, g \rangle_{\mathcal{L}^2(\mathcal{X}, \rho_{\mathcal{M}})} = \int_{\mathcal{X}} f(x)g(x) d\rho_{\mathcal{M}}(x)$ .

$$\|f_{\ell}\|_{\mathcal{L}^2(\mathcal{X}, \rho_{\mathcal{M}})} = \sqrt{\int_{\mathcal{X}} \left( \frac{\alpha(W^{\ell})}{p(W^{\ell})} \right)^2 (\phi(x, W^{\ell}))^2 d\rho_{\mathcal{M}}(x)},$$

Let  $E$  and  $F$  be the event defined as follows:

$$E = \left\{ \sup_{x \in \mathcal{M}} |\phi(x, W)| \leq M \right\}$$

$$F = \left\{ \left\| \frac{1}{m} \sum_{j=1}^m f_j - f \right\|_{\mathcal{L}^2(\mathcal{X}, \rho_{\mathcal{M}})} > t \right\}$$

Conditioned on  $E$  we have:

$$\|f_{\ell}\|_{\mathcal{L}^2(\mathcal{X}, \rho_{\mathcal{M}})} \leq CM.$$

$$\mathbb{P}(F) = \mathbb{P}(F|E) \mathbb{P}(E) + \mathbb{P}(F|E^c) \mathbb{P}(E^c) \leq \mathbb{P}(F|E) + \mathbb{P}(E^c).$$

Conditioned on the event  $E$ , we can apply McDiarmid inequality and we have:

$$\mathbb{P}(F|E) \leq \exp \left( -\frac{mt^2}{2M^2C^2} \right) = \delta_1$$

For  $\delta > 0$  set  $M = C_1 \sqrt{d_{\mathcal{M}} \log \left( \frac{\text{diam}(\mathcal{M}) \sqrt{d}}{\delta} \right) + \log(q+1)}$  applying Lemma 2 we have:

$$\mathbb{P}(E^c) \leq 1 - \delta - 2e^{-cd/4}$$

We have therefore with probability  $1 - \delta - 2e^{-cd/4} - \delta_1$ :

$$\left\| \frac{1}{m} \sum_{j=1}^m f_j - f \right\|_{\mathcal{L}^2(\mathcal{X}, \rho_{\mathcal{M}})} \leq CC_1 \sqrt{\frac{d_{\mathcal{M}} \log \left( \frac{\text{diam}(\mathcal{M}) \sqrt{d}}{\delta} \right) + \log(q+1)}{m}} \left( 1 + \sqrt{2 \log \left( \frac{1}{\delta_1} \right)} \right). \quad (9)$$

$\square$

The following Lemma shows how the approximation of functions in  $\mathcal{F}$ , by functions in  $\hat{\mathcal{F}}$ , transfers to the expected Risk:

**Lemma 4** (Bound on the Approximation Error). *Let  $f \in \mathcal{F}$ , fix  $\delta > 0$ . There exists a function  $\hat{f} \in \hat{\mathcal{F}}$ , such that:*

$$\mathcal{E}_V(\hat{f}) \leq \mathcal{E}_V(f) + LCC_1 \sqrt{\frac{d_{\mathcal{M}} \log\left(\frac{\text{diam}(\mathcal{M})\sqrt{d}}{\delta}\right) + \log(q+1)}{m}} \left(1 + \sqrt{2 \log\left(\frac{1}{\delta}\right)}\right)$$

with probability at least  $1 - 2\delta - 2e^{-cd/4}$ .

*Proof of Lemma 4.*  $\mathcal{E}_V(\hat{f}) - \mathcal{E}_V(f) \leq \int_{\mathcal{X}} |V(y\hat{f}(x)) - V(yf(x))| d\rho_{\mathcal{M}}(x) \leq L \int_{\mathcal{X}} |\hat{f}(x) - f(x)| d\rho_{\mathcal{M}}(x) \leq L \sqrt{\int_{\mathcal{X}} (\hat{f}(x) - f(x))^2 d\rho_{\mathcal{M}}(x)} = L \|\hat{f} - f\|_{\mathcal{L}^2(\mathcal{X}, \rho_{\mathcal{M}})}$ , where we used the Lipschitz condition and Jensen inequality. The rest of the proof follows from Lemma 3.  $\square$

We are now ready to prove Theorem 2.

*Proof of Theorem 2.* Let  $\hat{f}_N = \arg \min_{f \in \hat{\mathcal{F}}} \hat{\mathcal{E}}_V(f)$ ,  $\hat{f} = \arg \min_{f \in \hat{\mathcal{F}}} \mathcal{E}_V(f)$ ,  $f^* = \arg \min_{f \in \mathcal{F}} \mathcal{E}_V(f)$ .

$$\mathcal{E}_V(\hat{f}_N) - \min_{f \in \mathcal{F}} \mathcal{E}_V(f) = \underbrace{(\mathcal{E}_V(\hat{f}_N) - \mathcal{E}_V(\hat{f}))}_{\text{Statistical Error}} + \underbrace{(\mathcal{E}_V(\hat{f}) - \mathcal{E}_V(f^*))}_{\text{Approximation Error}}$$

**Bounding the statistical error.** The first term is the usual estimation or statistical error than we can bound as follows:

$$\begin{aligned} \mathcal{E}_V(\hat{f}_N) - \mathcal{E}_V(\hat{f}) &= (\mathcal{E}_V(\hat{f}_N) - \hat{\mathcal{E}}_V(\hat{f}_N)) + \underbrace{(\hat{\mathcal{E}}_V(\hat{f}_N) - \hat{\mathcal{E}}_V(\hat{f}))}_{\leq 0, \text{ by optimality of } \hat{f}_N} + (\hat{\mathcal{E}}_V(\hat{f}) - \mathcal{E}_V(\hat{f})) \\ &\leq 2 \sup_{f \in \hat{\mathcal{F}}} |\mathcal{E}_V(f) - \hat{\mathcal{E}}_V(f)|. \end{aligned}$$

Assume that the loss  $V : \mathbb{R} \rightarrow [0, 1]$ , when the data  $(x_i, y_i)$  or the random projections  $W^\ell$  change  $\sup_{f \in \hat{\mathcal{F}}} |\mathcal{E}_V(f) - \hat{\mathcal{E}}_V(f)|$ , can change by no more than  $\frac{2}{N}$  then by applying McDiarmids inequality [29] we have with a probability at least  $1 - \delta/2$

$$\sup_{f \in \hat{\mathcal{F}}} |\mathcal{E}_V(f) - \hat{\mathcal{E}}_V(f)| \leq \mathbb{E}_{x, W} \left( \sup_{f \in \hat{\mathcal{F}}} |\mathcal{E}_V(f) - \hat{\mathcal{E}}_V(f)| \right) + \sqrt{\frac{2 \log(2/\delta)}{N}}.$$

Now using the classical rademacher complexity type bounds [29], we have:

$$\mathbb{E}_{x, W} \sup_{f \in \hat{\mathcal{F}}} |\mathcal{E}_V(f) - \hat{\mathcal{E}}_V(f)| \leq 2L\mathcal{R}_N(\hat{\mathcal{F}}) + \frac{|V(0)|}{\sqrt{N}},$$

where  $\mathcal{R}_N(\hat{\mathcal{F}})$  is defined as follows:

$$\mathcal{R}_N(\hat{\mathcal{F}}) = \mathbb{E}_{x, W, \sigma} \left[ \sup_{f \in \hat{\mathcal{F}}} \left| \frac{1}{N} \sum_{i=1}^N \sigma_i f(x_i) \right| \right],$$

where  $\sigma_i$  are iid Rademacher variables  $\in \{-1, 1\}$ , such that  $\mathbb{P}(\sigma_i = 1) = \mathbb{P}(\sigma_i = -1) = \frac{1}{2}$ .

It is sufficient to bound the Rademacher complexity of the class  $\hat{\mathcal{F}}$ , where the expectation is taken over the randomness of the data and the random features:

$$\begin{aligned}
\mathcal{R}_N(\hat{\mathcal{F}}) &= \mathbb{E}_{x,W,\sigma} \left[ \sup_{f \in \hat{\mathcal{F}}} \left| \frac{1}{N} \sum_{i=1}^N \sigma_i f(x_i) \right| \right] = \mathbb{E}_{x,W,\sigma} \left[ \sup_{f \in \hat{\mathcal{F}}} \left| \frac{1}{N} \sum_{i=1}^N \sigma_i \left( \sum_{\ell=1}^m \alpha_\ell \phi(x_i, W^\ell) \right) \right| \right] \\
&= \mathbb{E}_{x,W,\sigma} \left[ \sup_{f \in \hat{\mathcal{F}}} \left| \frac{1}{N} \sum_{\ell=1}^m \alpha_\ell \sum_{i=1}^N \sigma_i \phi(x_i, W^\ell) \right| \right] \\
&\leq \mathbb{E}_{x,W,\sigma} \frac{1}{N} \|\alpha\|_\infty \sum_{\ell=1}^m \left| \sum_{i=1}^N \sigma_i \phi(x_i, W^\ell) \right| \quad \text{By Holder inequality: } \langle a, b \rangle \leq \|a\|_\infty \|b\|_1 \\
&\leq \frac{C}{mN} \mathbb{E}_{x,W} \sum_{\ell=1}^m \sqrt{\mathbb{E}_\sigma \left( \sum_{i=1}^N \sigma_i \phi(x_i, W^\ell) \right)^2} \quad \text{Jensen inequality, concavity of square root}
\end{aligned}$$

Note that  $\mathbb{E}(\sigma_i \sigma_j) = 0$ , for  $i \neq j$  it follows that:

$$\mathbb{E}_\sigma \left( \sum_{i=1}^N \sigma_i \phi(x_i, W^\ell) \right)^2 = \mathbb{E}_\sigma \sum_{i=1}^N \sum_{j=1}^N \sigma_i \sigma_j \phi(x_i, W^\ell) \phi(x_j, W^\ell) = \sum_{i=1}^N \phi^2(x_i, W^\ell).$$

Finally:

$$\begin{aligned}
\mathcal{R}_N(\hat{\mathcal{F}}) &\leq \frac{C}{mN} \sum_{\ell=1}^m \mathbb{E}_{x,W} \left( \sqrt{\sum_{i=1}^N \phi^2(x_i, W^\ell)} \right) \\
&= \frac{C}{N} \mathbb{E}_{x,W} \left( \sqrt{\sum_{i=1}^N \phi^2(x_i, W)} \right) \\
&\leq \frac{C}{N} \sqrt{\mathbb{E}_{x,W} \left( \sum_{i=1}^N \phi^2(x_i, W) \right)} \quad \text{By Jensen inequality} \\
&= \frac{C}{N} \sqrt{N \mathbb{E}_{x,W} \phi^2(x, W)} \\
&\leq \frac{C}{\sqrt{N}} \sqrt{\mathbb{E}_x(K(x, x))}.
\end{aligned}$$

Recall that for  $x \in \mathbb{S}^{d-1}$ ,  $K(x, x) = \sigma^2(q) \|x\|^2 \kappa_q(x, x) = \sigma^2(q)$ . Hence:

$$\mathcal{R}_m(\hat{\mathcal{F}}) \leq C \sqrt{\frac{\sigma^2(q)}{N}},$$

hence we have with probability  $1 - \delta/2$ , on the choice of random data and random projections:

$$\mathcal{E}_V(\hat{f}_N) - \mathcal{E}_V(\hat{f}) \leq 4LC \sqrt{\frac{\sigma^2(q)}{N}} + \frac{2|V(0)|}{\sqrt{N}} + 2\sqrt{\frac{2 \log(2/\delta)}{N}}. \quad (10)$$

**Bounding the Approximation Error.** Let  $\hat{f}^*$ , the function defined in Lemma 3, that approximates  $f^*$  in  $\hat{\mathcal{F}}$ . By Lemma 4 we know that:

$$\mathcal{E}_V(\hat{f}^*) \leq \mathcal{E}_V(f^*) + LCC_1 \sqrt{\frac{d_{\mathcal{M}} \log \left( \frac{\text{diam}(\mathcal{M}) \sqrt{d}}{\delta} \right) + \log(q+1)}{m}} \left( 1 + \sqrt{2 \log \left( \frac{1}{\delta} \right)} \right)$$

with probability  $1 - 2\delta - 2e^{-cd/4}$ , on the choice of the random projections. By optimality of  $\hat{f} \in \tilde{\mathcal{F}}$ , we have with at least the same probability  $1 - 2\delta - 2e^{-cd/4}$

$$\mathcal{E}_V(\hat{f}) \leq \mathcal{E}_V(\hat{f}^*) \leq \mathcal{E}_V(f^*) + LCC_1 \sqrt{\frac{d_{\mathcal{M}} \log \left( \frac{\text{diam}(\mathcal{M}) \sqrt{d}}{\delta} \right) + \log(q+1)}{m}} \left( 1 + \sqrt{2 \log \left( \frac{1}{\delta} \right)} \right)$$

Hence by a union bound with probability  $1 - 3\delta - 2e^{-cd/4}$ , on the training set and the random projections:

$$\begin{aligned} \mathcal{E}_V(\hat{f}_N) - \min_{f \in \mathcal{F}} \mathcal{E}_V(\hat{f}) &\leq 4LC \sqrt{\frac{\sigma^2(q)}{N}} + \frac{2|V(0)|}{\sqrt{N}} + 2\sqrt{\frac{2\log(1/\delta)}{N}} \\ &\quad + LCC_1 \sqrt{\frac{d_{\mathcal{M}} \log\left(\frac{\text{diam}(\mathcal{M})\sqrt{d}}{\delta}\right) + \log(q+1)}{m}} \left(1 + \sqrt{2\log\left(\frac{1}{\delta}\right)}\right). \end{aligned}$$

□

## References

- [1] V. N. Vapnik, *Statistical learning theory*. A Wiley-Interscience Publication, 1998.
- [2] B. Scholkopf, A. Smola, and K.-R. Müller, “Kernel principal component analysis,” in *advances in Kernel learning*, pp. 327–352, MIT Press, 1999.
- [3] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” in, vol. 86, pp. 2278–2324, 1998.
- [4] A. Rahimi and B. Recht, “Random features for large-scale kernel machines,” *NIPS*, 2007.
- [5] P. Huang, H. Avron, T. Sainath, V. Sindhvani, and B. Ramabhadran, “Kernel methods match deep neural networks on timit,” *In ICASSP*, 2014.
- [6] G. Wahba, *Spline models for observational data*, vol. 59 of *CBMS-NSF*. Philadelphia, PA: SIAM, 1990.
- [7] D. Lopez-Paz, S. Sra, A. J. Smola, Z. Ghahramani, and B. Scholkopf, “Randomized nonlinear component analysis,” *ICML*, 2014.
- [8] I. Goodfellow, D. Warde-Farley, M. Mirza, A. C. Courville, and Y. Bengio, “Maxout networks,” *ICML*, 2013.
- [9] A. Frieze and M. Jerrum, “Improved approximation algorithms for max k-cut and max bisection,” 1995.
- [10] R. Galambos, “The asymptotic theory of extreme order statistics,” *John Wiley and sons*, 1940.
- [11] P. Indyk, “Algorithmic applications of low-distortion geometric embeddings,” in *FOCS*, pp. 10–33, IEEE Computer Society, 2001.
- [12] J. Heinonen, *Lectures on Analysis on Metric Spaces*. Springer, Springer, 2001.
- [13] A. Rahimi and B. Recht, “Uniform approximation of functions with random bases,” in *Proceedings of the 46th Annual Allerton Conference*, 2008.
- [14] F. R. Bach, “On the equivalence between quadrature rules and random features,” *CoRR*, vol. abs/1502.06800, 2015.
- [15] R. J. Durrant and A. Kaban, “Sharp generalization error bounds for randomly-projected classifiers,” in *ICML*, pp. 693–701, JMLR W&CP volume 28 (3), 2013.
- [16] W. B. Johnson and J. Lindenstrauss, “Extensions of lipschitz mappings into a hilbert space,” *Conference in modern analysis and probability*, vol. Contemp. Math., 26, Amer. Math. Soc., Providence, RI, p. 189206, 1984.
- [17] Y. Bartal, B. Recht, and L. J. Schulman, “Dimensionality reduction: Beyond the johnson-lindenstrauss bound,” in *SODA*, 2011.
- [18] B. Dai, B. Xie, N. He, Y. Liang, A. Raj, M. Balcan, and L. Song, “Scalable kernel methods via doubly stochastic gradients,” in *NIPS*, pp. 3041–3049, 2014.
- [19] A. J. Smola and B. Schlkopf, “Sparse greedy matrix approximation for machine learning,” in *ICML*, 2000.
- [20] C. Williams and M. Seeger, “Using the nystrom method to speed up kernel machines,” in *NIPS 13*, pp. 682–688, MIT Press, 2001.

- [21] S. Fine, K. Scheinberg, N. Cristianini, J. Shawe-taylor, and B. Williamson, “Efficient svm training using low-rank kernel representations,” *JMLR*, vol. 2, pp. 243–264, 2001.
- [22] S. T. Roweis and L. K. Saul, “Nonlinear dimensionality reduction by locally linear embedding,” *Science*, 2000.
- [23] A. Magnani and S. P. Boyd, “Convex piecewise-linear fitting,” 2006.
- [24] A. Rahman Mohamed, T. N. Sainath, G. E. Dahl, B. Ramabhadran, G. E. Hinton, and M. A. Picheny, “Deep belief networks using discriminative features for phone recognition.,” in *ICASSP*, 2011.
- [25] G. E. Dahl, T. N. Sainath, and G. E. Hinton “Improving deep neural networks for LVCSR using rectified linear units and dropout ,” *ICASSP*,.
- [26] A. Frieze and M. Jerrum, “Improved approximation algorithms for max k-cut and max bisection,” 1995.
- [27] J. Heinonen, *Lectures on Analysis on Metric Spaces*. Springer, Springer, 2001.
- [28] R. Vershynin, “Introduction to the non-asymptotic analysis of random matrices,” *Compressed Sensing: Theory and Applications*, Y. Eldar and G. Kutyniok, Eds. Cambridge University Press., 2011.
- [29] P. L. Bartlett and S. Mendelson, “Rademacher and gaussian complexities: Risk bounds and structural results,” *J. Mach. Learn. Res.*, vol. 3, pp. 463–482, Mar. 2003.
- [30] A. Tacchetti, P. K. Mallapragada, M. Santoro, and L. Rosasco, “Gurls: a least squares library for supervised learning,” *CoRR*, vol. abs/1303.0934, 2013.